A Context Sensitive Grammer generating the set of all primitive words(Algebras, Languages, Computations and their Applications)

Kunimochi, Yoshiyuki

数理解析研究所講究録
第 1562 巻 2007 年 143-145

A Context Sensitive Grammer generating the set of all primitive words *

静岡理工科大学 國持良行 (Yoshiyuki Kunimochi)

Department of Computer Science, Shizuoka Institute of Science and Technology
Toyosawa 2200-2, Fukuroi-shi, Shizuoka 437-8555, Japan

Abstract

It is still an open problem whether the set Q of all primitive words is a context free language or not. There are many related works to this problem. It is known that Q is not a deterministic context free language but it is a (deterministic) context sensitive language[1]. However, a context sensitive grammer generating Q have not explicitly given so far. We give such a context sensitive grammer here.

1 Introduction

Let X be a finite nonempty set of *letters*, called an alphabet and let X^* be the free monoid generated by X under the operation of catenation. An element of X^* is called a *word* over X. The identity of X^* , denoted by ε, is called the *empty word* and $X^* \setminus \{\varepsilon\}$ is denoted by X^+. The catenation of two words x and y is denoted by xy. A word $x \in X^+$ is primitive if $y = f^n$ for some $f \in X^+$ implies $n = 1$, where $f^n = \overbrace{ff\cdots f}^{n}$.

A phrase-structure grammar G is a quadruple $G = (V, X, S, P)$ where V(resp.X) is a finite nonempty set of nonterminal (resp.terminal) symbols, $S \in V$ is the initial symbol, and $P \subset (V \cup X)^* V (V \cup X)^* \times (V \cup X)^*$ is a finite nonempty set of production rules. A production rule $(l, r) \in P$ is often written in the form $l \rightarrow r$. A relation $\rightarrow$ is extended on $(V \cup X)^*$ and defined by $xly \rightarrow xry$ where $(l, r) \in P$ and $x, y \in (V \cup X)^*$. The transitive closure and reflective and transitive closure of $\rightarrow$ is denoted by $\rightarrow^+$ and $\rightarrow^*$, respectively. The language generated by G is defined by $\{w \in X^* | S \rightarrow^* w\}$ and denoted by $L(G)$. The number of nonterminal symbols and terminal symbols occured in $\alpha \in (V \cup X)^*$ is denoted by $|\alpha|$.

A phrase-structure grammar $G = (V, X, S, P)$ is said a context sensitive grammer (CSG for short) if every production rule $l \rightarrow r$ satisfies $|l| \leq |r|$. Then a languege generated by a CSG is called a context sensitive language (CSL for short).

2 CSG generating the set of all primitive words over X

At first, we give the phase-structure grammer G which generate the set Q of all primitive words. G is very close to be a CSL. After that, we transform the grammer into the context sensitive one by new special nonterminal symbols.

Lemma 2.1 Let $Q \subset X^+$ be the set of all primitive words. There is some phase-structure grammer G which generates Q and whose maximum length on computation is $n + 6$, for each primitive word of the length n.

(Sketch of Proof) We define the phrase-structure grammer $G = (V, X, S, P)$ as follows: The set V of all nonterminal symbols is defined by

$$V = \{S, S_1, L, \overleftarrow{L}, R, \sigma_0, \sigma_1, \overleftarrow{\sigma}, \overrightarrow{\sigma_a}, \overrightarrow{\sigma_R}, \overline{\sigma}, \tau_0, \tau, \$, \& | a \in X\}.$$

The initial symbol is S. The production rules $P = \bigcup_{i=1}^{8} P_i$:

(Produce an initial configuration)
$P_1 = \{S \rightarrow \$LRS_1\sigma_0\tau_0\&, S_1 \rightarrow aS_1, S_1 \rightarrow a | a \in X\}$,

* This is an abstract and the paper will appear elsewhere.

(get ready to check mode again)
$P_2 = \{a\sigma_0 \to \sigma_0 a, R\sigma_0 \to \sigma_0 R, L\sigma_0 \to \overleftarrow{L}\sigma_1, \sigma_1 a \to a\sigma_1, a\tau_0 \to \tau_0 a | a \in X\}$.

(Move R one letter to the right and start to check the multiplicity)
$P_3 = \{\sigma_1 R\tau_0 a \to aR\overleftarrow{\sigma}\tau | a \in X\}$,

(Scanner σ goes to the left and memorize the next letter)
$P_4 = \{a\overleftarrow{\sigma} \to \overleftarrow{\sigma}a, R\overleftarrow{\sigma} \to \overleftarrow{\sigma}R, L\overleftarrow{\sigma}a \to aL\overrightarrow{\sigma}_a, L\overleftarrow{\sigma}R \to \overleftarrow{L}R\overrightarrow{\sigma}_R | a \in X\}$,

(Scanner $\overrightarrow{\sigma}_a$ goes to the right and check the letter)
$P_5 = \{\overrightarrow{\sigma}_a b \to b\overrightarrow{\sigma}_a, \overrightarrow{\sigma}_a R \to R\overrightarrow{\sigma}_a, \overrightarrow{\sigma}_a \tau a \to \overleftarrow{\sigma}a\tau, \overrightarrow{\sigma}_a \tau c \to \sigma_0\tau_0 c, \overrightarrow{\sigma}_a \tau\& \to \sigma_0\tau_0\& | a, b, c \in X, a \neq c\}$,

(Scanner $\overrightarrow{\sigma}_R$ goes to the right and check the multiplicity)
$P_6 = \{\overrightarrow{\sigma}_R a \to a\overrightarrow{\sigma}_R, \overrightarrow{\sigma}_R R \to R\overrightarrow{\sigma}_R, \overrightarrow{\sigma}_R \tau a \to \overleftarrow{\sigma}\tau a | a \in X\}$,

(Movement of the Left-mergin $\overleftarrow{L}$)
$P_7 = \{a\overleftarrow{L} \to \overleftarrow{L}a, \$\overleftarrow{L} \to \$L | a \in X\}$,

(Clean up all 6 non-terminal symbols)
$P_8 = \{\sigma_1 R\tau_0\& \to E, aE \to Ea, \$LE \to \varepsilon | a \in X\}$,

Then the grammer G generates Q. It is easily shown that when G generates a primitive word x, the length of each configuration on its derivation is equal to or shorter than $|x| + 6$.

EXAMPLE 2.1 These are examples of derivations by the grammer G.

(1)

S
$\to^* \$LRa\sigma_0\tau_0\&$
$\to^* \$L\sigma_1 R\tau_0 a\&$
$\to \$LaR\overleftarrow{\sigma}\tau\&$ start to check
$\to^* \$aLR\overrightarrow{\sigma}_a\tau\&$
$\to \$aLR\sigma_0\tau_0\&$
$\to^* \$La\sigma_1 R\tau_0\&$ success
$\to^* a$.

(2)

S
$\to^* \$LRaa\sigma_0\tau_0\&$
$\to^* \$LaR\overleftarrow{\sigma}\tau a\&$ start to check
$\to^* \$aLR\overrightarrow{\sigma}_a\tau a\&$
$\to \$aLR\overleftarrow{\sigma}a\tau\&$
$\to^* \$aLRa\overrightarrow{\sigma}_R\tau\&$. fail

This derivation cannot go any longer.

(3)

S
$\to^* \$LRab\sigma_0\tau_0\&$
$\to^* \$LaR\overleftarrow{\sigma}\tau b\&$ start to check
$\to^* \$aLR\overrightarrow{\sigma}_a\tau b\&$
$\to \$aLR\sigma_0\tau_0 b\&$
$\to^* \$La\sigma_1 R\tau_0 b\&$
$\to \$LabR\overleftarrow{\sigma}\tau\&$ start to check
$\to^* \$aLbR\overrightarrow{\sigma}_a\tau\&$
$\to \$aLbR\sigma_0\tau_0\&$
$\to^* \$Lab\sigma_1 R\tau_0\&$ success
$\to^* ab$.

(4)

$$
\begin{aligned}
& S \\
& \to^* \$LRaab\sigma_0\tau_0\& \\
& \to^* \$LaR\overleftarrow{\sigma}\tau ab\& && \text{start to check} \\
& \to^* \$aLR\overrightarrow{\sigma}_a\tau ab\& \\
& \to \$aLR\overleftarrow{\sigma}a\tau b\& \\
& \to^* \$aLRa\overrightarrow{\sigma}_R\tau b\& \\
& \to \$aLRa\sigma_0\tau_0 b\& \\
& \to^* \$LaaR\overleftarrow{\sigma}\tau b\& && \text{start to check} \\
& \to^* \$aLaR\overrightarrow{\sigma}_a\tau b\& \\
& \to \$aLaR\sigma_0\tau_0 b\& \\
& \to^* \$LaabR\overleftarrow{\sigma}\tau\& && \text{start to check} \\
& \to^* \$aLabR\overrightarrow{\sigma}_a\tau\& \\
& \to \$aLabR\sigma_0\tau_0\& \\
& \to^* \$Laab\sigma_1 R\tau_0\& && \text{success} \\
& \to^* aab.
\end{aligned}
$$

Lemma 2.2 Let $Q \subset X^+$ be the set of all primitive words. Then there is a CSG G which generates Q.

(Sketch of Proof) We define the following CSG $G = (V, X, S, P)$ as follows. Let V_0 be the set of the foundamental nonterminal symbols, V_1 be the set of the contextual nonterminal symbols.

$$
\begin{aligned}
& V_0 = \{L, \overleftarrow{L}, R, \sigma_0, \sigma_1, \overleftarrow{\sigma}, \overrightarrow{\sigma}_a, \overrightarrow{\sigma}_R, \bar{\sigma}, \tau_0, \tau, \$, \& | a \in X\}, \\
& V_1 = \{[\alpha] | \alpha \in (V_0 \cup X)^3 \cup (V_0 \cup X)^5\}, \\
& V = \{S, S_1\} \cup V_0 \cup V_1,
\end{aligned}
$$

where S is the initial symbol. The production rules $P = \cup_{i=1}^{9} P_i$:

(Produce an initial configuration)
$P_1 = \{S \to a, S \to ab, S \to [\$LR]S_1[ac\sigma_0\tau_0\&], S_1 \to a, S_1 \to aS_1 | a, b, c \in X, a \neq b\}$

Each rule from P_2 to P_7 is the same production rule as that in Lemma 2.1, respectively. Note that each rules $l \to r$ in P_2 to P_7 has the property of $|l| = |r|$.

(Rules for the contextural nonterminal symbols)
$P_8 = \{[\gamma\alpha_1]\alpha_2 \to [\gamma\beta_1]\beta_2, \alpha_1[\alpha_2\gamma] \to \beta_1[\beta_2\gamma], [\gamma\alpha\delta] \to [\gamma\beta\delta], |\alpha \to \beta \in \cup_{i=2}^{7} P_i, \alpha = \alpha_1\alpha_2, \beta = \beta_1\beta_2, \gamma, \delta \in (V_0 \cup X)^*, |\alpha_1| = |\beta_1| > 0, |\alpha_2| = |\beta_2| > 0\}$

(Eliminate all 6 nonterminal symbols)
$P_9 = \{[\$La] \to a, [a\sigma_1 R\tau_0\&] \to a | a \in X\}$

References

[1] Dömösi,P., Horváth, S., Ito,M.,: Formal languages and primitive words, Publ. Math., Debrecen, 42, (1993) pp.315–321.

[2] Dömösi,P., Horváth, S., Ito,M., Kászonyi, L., Katsura,M.,: Formal languages consisting of primitive words, LNCS, 710, (1993), pp.194–203.

[3] Ito,M., Katsura, M., Shyr,H., Yu,S.S.: Automata accepting primitive words, Semigroup Forum, 37, (1988), pp.45–52.

[4] Ito,M., Katsura, M.: Context-free languages consisting of non-primitive words, Int. Journal of Comp. Math., 40, (1991), pp.157–167.